js中的隐式转换

js中的不同的数据类型之间的比较转换规则如下：

**1. 对象和布尔值比较**

**对象和布尔值进行比较时，对象先转换为字符串，然后再转换为数字，布尔值直接转换为数字**

[] == true; //false []转换为字符串'',然后转换为数字0,true转换为数字1，所以为false

**2. 对象和字符串比较**

**对象和字符串进行比较时，对象转换为字符串，然后两者进行比较。**

[1,2,3] == '1,2,3' // true [1,2,3]转化为'1,2,3'，然后和'1,2,3'， so结果为true;

**3. 对象和数字比较**

**对象和数字进行比较时，对象先转换为字符串，然后转换为数字，再和数字进行比较。**

[1] == 1; // true `对象先转换为字符串再转换为数字，二者再比较 [1] => '1' => 1 所以结果为true

**4. 字符串和数字比较**

**字符串和数字进行比较时，字符串转换成数字，二者再比较。**

'1' == 1 // true

**5. 字符串和布尔值比较**

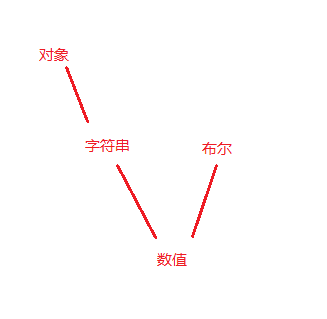
**字符串和布尔值进行比较时，二者全部转换成数值再比较。**

'1' == true; // true

**6. 布尔值和数字比较**

**布尔值和数字进行比较时，布尔转换为数字，二者比较。**

true == 1 // true

许多刚接触js的童鞋看到这么多的转换规则就懵圈了，其实规律很简单，大家可以记下边这个图  


如图，任意两种类型比较时，如果不是同一个类型比较的话，则按如图方式进行相应类型转换，如对象和布尔比较的话，对象 => 字符串 => 数值 布尔值 => 数值。  
另外，我们来看下一些需要"特别照顾"的。

来看一个有趣的题

[] == false;

![] == false;

这两个的结果都是true,第一个是，对象 => 字符串 => 数值0 false转换为数字0,这个是true应该没问题，  
第二个前边多了个!，则直接转换为布尔值再取反，转换为布尔值时，**空字符串(''),NaN,0，null,undefined**这几个外返回的都是true, 所以! []这个[] => true 取反为false,所以[] == false为true。

**还有一些需要记住的，像：**

undefined == null //true undefined和null 比较返回true，二者和其他值比较返回false

Number(null) //0